REMARKS

I. STATUS OF THE CLAIMS

Claims 1-3, 5-7 and 19-21 are canceled herein.

In view of the above it is respectfully submitted that claims 8-18 are currently pending.

II. REJECTION OF CLAIMS 8-18 UNDER 35 USC 103 AS BEING UNPATENTABLE OVER OWENS (US PATENT NO. 6,529,915) IN VIEW OF SHEN (US PATENT NO. 5,937,410)

The present invention as recited, for example, in claim 8, relates to a method comprising (a) selecting object-oriented data in an object model by a user, the selected data being stored in a database by the object model; (b) automatically extracting the selected data from the database by directly extracting the selected data from the object model using an object query language corresponding to the object model; (c) automatically building tables for the extracted data in accordance with metadata for the extracted data, the tables being tables for a target relational database; and (d) automatically inserting the extracted data into the tables using a query language corresponding to the tables and which is different from the object query language.

See, for example, FIG. 4 of the present application. For example, in operation 62 in FIG. 4, a user selects object-oriented data in an object model by a user, the selected data being stored in a database by the object model. In operation 70 in FIG. 4, the selected data is automatically extracted from the database by directly extracting the selected data from the object model using an object query language corresponding to the object model. In operation 71 in FIG. 4, tables for the extracted data are automatically built in accordance with metadata for the extracted data, the tables being tables for a target relational database. In operations 76 and 80 in FIG. 4, the extracted data is automatically inserted into the tables using a query language corresponding to the tables and which is different from the object query language.

See also, for example, FIG. 3 of the present application. For example, FIG. 3 discloses an object model 22 that stores data in a database 20. As shown in FIG. 3, data stored in database 20 is automatically extracted from the database by directly extracting the data from object model 22 using an object query language (OQL) corresponding to the object model. For example, in FIG. 3, please see the line from element 54 to object model 22, with the notation "OQL". This operation in FIG. 3 can be compared to the prior art in FIG. 1 of the present application, where an export engine 34 exports data from database 20, not from object model

22. See, for example, paragraph [0008] of the specification.

Owens discloses a system for accessing a relational database through an object-oriented querying interface.

FIG. 3 of Owens shows an object-oriented application 101 that accesses data from a relational database (RDMS) 107. An object server 105 translates object-oriented requests into relational database requests (SQL). See, for example, column 5, lines 45-46, of Owens.

Object server 105 stores and retrieves data from various tables stored in relational database 107 according to a hierarchical tree that maps data encapsulated within objects to table locations in relational database 107. See, for example, column 1, lines 26-30; and column 5, lines 54-56, of Owens.

As shown in FIG. 7 of Owens, to define and store an object, a user defines a class for the object . See for example, operation 301, and column 9, lines 2-3, of Owens. An object-oriented application then analyzes class definitions and generates a hierarchical tree for parsing. See, for example, column 9, lines 7-8, of Owens. The hierarchical tree is parsed in order to map from the object-oriented scheme to the relational database scheme. See, for example, column 9, lines 11-13, of Owens.

The object server 105 references container objects and generates SQL calls to the relational database 107 to store data in the appropriate relational database tables. See, for example, column 9, lines 58-60, of Owens.

FIGS. 13, 16A and 16B of Owens show a process of querying the relational database 107 for information about objects of interest through an object-oriented interface. More specifically, as shown by these figures, a user specifies a query to an object-oriented system. An object-oriented application then "fills in the query container object" by inserting a query template, args array and results array into the query container object. The query template is an SQL-like query that includes object-oriented information. See, for example, column 13, lines 20-25, of Owens. SQL is then generated from the container objects. The object server 105 then issues the query to the relational database 107. See operation 759 in Owens. The objects server 105 then receives the requested information from the relational database 107.

In summary, Owens directly accesses a relational database 107 via an object-oriented application, to retrieve and store data in the relational database 107. For example, in FIG. 3 of Owens, object server 105 directly accesses relational database 107 using a relational database query language, such as SQL.

Therefore, Owens is similar to the prior art as disclosed in FIG. 1 of the present application, where a user interface 28 allows an end-user to view and access an object model 22, and the object model accesses a relational database 20 in accordance with user requests.

The above-described operation in Owens is significantly different than the present invention as recited, for example, in claim 8, where selected data is **directly extracted from the object model using an object query language corresponding to the object model**.

Moreover, as indicated above, Owens uses a hierarchical tree to map data encapsulated within objects to table locations in relational database 107. These hierarchical trees and maps of Owens must be predetermined and stored in advance of being used. For example, as shown in FIG. 7 of Owens, a hierarchical tree is generated at the time an object is defined and relational database tables are created. This operation of Owens is significantly different than the present invention as recited, for example, in claim 8, where **tables are automatically built for extracted data in accordance with metadata for the extracted data**.

The Examiner asserts that FIG. 13, operations 605-611, of Owens, teaches automatic building of tables for the extracted data in accordance with metadata for the extracted data, the tables being tables for a target relational database. However, it is respectfully submitted that these portions of Owens do not disclose this feature. Instead, these portions of Owens disclose that a query is performed. More specifically, these portions of Owens disclose that an object-oriented inquiry is received from a user, the query is translated into SQL, an SQL query is then performed against a relational (RDBMS) database, and the results are put into a container object in memory. A container object is shown in FIGS. 5A/5B of Owens, and is used to store an object in transient memory.

The Examiner also asserts that column 9, lines 14-19, of Owens, teach automatic building of tables for the extracted data in accordance with metadata for the extracted data, the tables being tables for a target relational database. However, it is respectfully submitted that this portion of Owens does not disclose this feature. Instead, this portion of Owens describes operation 305 in FIG. 7 of Owens, which is related to defining and storing an object when initially defining an object. The operations in FIG. 7 of Owens include the user defining a class for the object, generating a tree for parsing, and creating the relational tables. No data is involved in these operations. These operations in Owens are significantly different than the present invention as recited, for example, in claim 8, where tables are automatically built for extracted data **in accordance with metadata for the extracted data**.

Therefore, it is respectfully submitted that the present invention as recited, for example, in claim 8, is significantly different than that disclosed in Owens.

On page 4 of the Office Action, the Examiner concedes that Owens does not explicitly teach automatically inserting the extracted data into the tables using a query language corresponding to the tables and which is different from the object query language. Instead, the Examiner asserts that this operation is disclosed in Shin.

Shen discloses the conversion of object oriented models into a database.

However, in Shen, **the actual object model is converted**. For example, in FIG. 1 of Shen, data input 14 is a file generated from a computer drawing program, and is indicative of identified relationships among database objects. See, for example, the Abstract, and column 3, lines 5-26, of Shen. Further, column 4, line 35, through column 5, line 28, of Shen, describes how the input data 14 is used to convert the object model into a database. See also column 1, lines 36-37, of Shen, indicating that a graphical object oriented model is transformed into a PDM (product data manager) scheme.

Therefore, in Shen, the actual object model is converted. This is different than the present invention as recited, for example, in claim 8.

Therefore, it is respectfully submitted that none of the references, taken individually or in combination, discloses or suggests the present invention as recited, for example, in claim 8.

Although the above comments are specifically directed to claim 8, it is respectfully submitted that the comments would be helpful in understanding various differences of various other claims over the cited references.

III. CONCLUSION

In view of the above, it is respectfully submitted that the application is in condition for allowance, and a Notice of Allowance is earnestly solicited.

If any further fees are required, please charge such fees to our Deposit Account No. 19-3935.

Respectfully submitted,
STAAS & HALSEY LLP

Date: March 1, 2006

By:
Paul I. Kravetz
Registration No. 35,230

1201 New York Avenue, NW, Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501